



Artificial Software Diversification for WebAssembly

JAVIER CABRERA-ARTEAGA

Doctoral Thesis
Supervised by
Benoit Baudry and Martin Monperrus
Stockholm, Sweden, 2023

TRITA-EECS-AVL-2020:4
ISBN 100-

KTH Royal Institute of Technology
School of Electrical Engineering and Computer Science
Division of Software and Computer Systems
SE-10044 Stockholm
Sweden

Akademisk avhandling som med tillstånd av Kungl Tekniska högskolan framlägges
till offentlig granskning för avläggande av Teknologie doktorexamen i elektroteknik
i .

© Javier Cabrera-Arteaga , date

Tryck: Universitetsservice US AB

Abstract

[1]

Keywords: Lorem, Ipsum, Dolor, Sit, Amet

Sammanfattning

[1]

LIST OF PAPERS

1. *WebAssembly Diversification for Malware Evasion*
Javier Cabrera-Arteaga, Tim Toady, Martin Monperrus, Benoit Baudry
Computers & Security, Volume 131, 2023
<https://www.sciencedirect.com/science/article/pii/S0167404823002067>
2. *Wasm-mutate: Fast and Effective Binary Diversification for WebAssembly*
Javier Cabrera-Arteaga, Nicholas Fitzgerald, Martin Monperrus, Benoit Baudry
3. *Multi-Variant Execution at the Edge*
Javier Cabrera-Arteaga, Pierre Laperdrix, Martin Monperrus, Benoit Baudry
Conference on Computer and Communications Security (CCS 2022), Moving Target Defense (MTD)
<https://dl.acm.org/doi/abs/10.1145/3560828.3564007>
4. *CROW: Code Diversification for WebAssembly*
Javier Cabrera-Arteaga, Orestis Floros, Oscar Vera-Pérez, Benoit Baudry, Martin Monperrus
Network and Distributed System Security Symposium (NDSS 2021), MADWeb
<https://doi.org/10.14722/madweb.2021.23004>
5. *Superoptimization of WebAssembly Bytecode*
Javier Cabrera-Arteaga, Shrinish Donde, Jian Gu, Orestis Floros, Lucas Satabin, Benoit Baudry, Martin Monperrus
Conference Companion of the 4th International Conference on Art, Science, and Engineering of Programming (Programming 2021), MoreVMs
<https://doi.org/10.1145/3397537.3397567>
6. *Scalable Comparison of JavaScript V8 Bytecode Traces*
Javier Cabrera-Arteaga, Martin Monperrus, Benoit Baudry
11th ACM SIGPLAN International Workshop on Virtual Machines and Intermediate Languages (SPLASH 2019)
<https://doi.org/10.1145/3358504.3361228>

ACKNOWLEDGEMENT

ACRONYMS

List of commonly used acronyms:

Wasm WebAssembly

Contents

List of Papers	iii
Acknowledgement	v
Acronyms	vii
Contents	1
I Thesis	3
1 Introduction	5
1.1 Background	5
1.2 Problem statement	5
1.3 Automatic Software diversification requirements	5
1.4 List of contributions	5
1.5 Summary of research papers	6
1.6 Thesis outline	6
2 Background and state of the art	7
2.1 WebAssembly	7
2.1.2 Generating WebAssembly programs	7
2.1.3 WebAssembly's binary format	9
2.1.4 WebAssembly's runtime structure	11
2.1.5 WebAssembly's control flow	13
2.1.6 WebAssembly's ecosystem	14
2.1.7 WebAssembly opportunities	17
2.2 Software diversification	18
2.2.2 Generating Software Diversification	18
2.2.3 Variants generation	18
2.2.4 Variants equivalence	18

2.3	Exploiting Software Diversification	18
2.3.2	Defensive Diversification	18
2.3.3	Offensive Diversification	18
3	Automatic Software Diversification for WebAssembly	19
3.1	CROW: Code Randomization of WebAssembly	20
3.1.2	Variants' generation	20
3.1.3	Constant inferring	22
3.1.4	Combining replacements	23
3.1.5	CROW instantiation	23
3.2	MEWE: Multi-variant Execution for WebAssembly	25
3.2.2	Multivariant generation	25
3.3	WASM-MUTATE: Fast and Effective Binary for WebAssembly	28
3.3.2	WebAssembly Rewriting Rules	29
3.3.3	Extending peephole meta-rules with custom operators	33
3.3.4	E-graphs	35
3.3.5	Random e-graph traversal for variants generation	35
3.3.6	WASM-MUTATE instantiation	37
3.4	Comparing CROW, MEWE, and WASM-MUTATE	39
3.4.2	Technology and approach	39
3.4.3	Strength of the generated variants	40
3.4.4	Security guarantees	40
3.5	Conclusions	42
4	Exploiting Software Diversification for WebAssembly	43
4.1	WASM-MUTATE WebAssembly malware evasion	43
4.1.2	Objective	43
4.1.3	Approach	43
4.1.4	Results	43
4.2	CROW: Automatic testing and fuzzing of WebAssembly consumers	43
4.2.2	Objective	43
4.2.3	Approach	44
4.2.4	Results	44
4.3	MEWE: Multivariant execution at the Edge	44
4.3.2	Threat model	44
4.3.3	Approach	44
4.3.4	Results	44
4.4	WASM-MUTATE: Speculative Side-channel protection	44
4.4.2	Threat model	44
4.4.3	Approach	44
4.4.4	Results	45

CONTENTS	3
----------	---

5 Conclusions and Future Work	47
5.1 Summary of technical contributions	47
5.2 Summary of empirical findings.	47
5.3 Summary of empirical findings.	47
5.4 Future Work	47
II Included papers	49
Superoptimization of WebAssembly Bytecode	53
CROW: Code Diversification for WebAssembly	55
Multi-Variant Execution at the Edge	57
WebAssembly Diversification for Malware Evasion	59
Wasm-mutate: Fast and Effective Binary Diversification for WebAssembly	61
Scalable Comparison of JavaScript V8 Bytecode Traces	63

Part I

Thesis

